浅析一类最小割问题

湖南师大附中 彭天翼

摘 要

在信息学竞赛中，网络流最小割的问题层出不穷，建模的思路往往十分巧妙。本文通过对一类最小割问题的研究，总结出了该类问题的通常解法，希望能够抛砖引玉，帮助大家对最小割有更深的理解。

目 录

1 预备知识

在阅读本文前，作者希望你能够做到：

1. 熟知网络流的定义以及最大流最小割定理。
2. 熟练网络流算法。
3. 最好能够阅读论文《胡伯涛：最小割模型在信息学中的应用》，它将对你理解最小割有莫大的帮助。

2 从一个简单题说起[poj 3469]

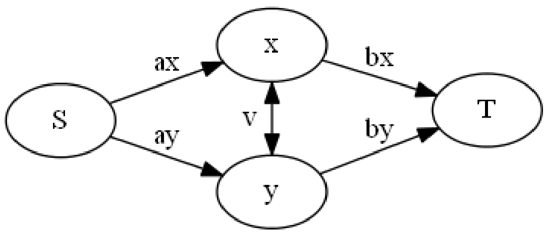
2.1 题目描述

有N个任务，每个任务可以在机器A或者机器B上完成,花费分别为a_i或者b_i。有m对关系，每对关系为二元组(x,y)，表示第x个任务和第y个任务如果不在相同的机器上完成，则增加v的花费。问如何安排完成每个任务的机器，使得总花费最小。

2.2 题目分析

本题的关键字在："如果两者不相同，则增加v的花费"。对于此类问题，有一个堪称经典的最小割建模方式：

新建源S，汇T，每个任务建一个点，标号1-N。S向每个任务连边，边权（容量）为ai，每个任务向T连边，边权为bi。对于每对二元组(x,y)，将x与y之间连双向边，边权为v。



首先来论述一下这种做法的正确性。考虑每个点，要么将其与S相连的边割掉（表示在机器A上完成，花费a_i），要么将其与T相连的边割掉（表示在机

器B上完成，花费b_i）。再考虑每一对有连边关系的点(x,y)，分下面四种情况讨论：

- 如果两个任务同时在A完成，需要割掉与S相连的边，花费ax+ay
- 如果两个任务同时在B完成，需要割掉与T相连的边，花费bx+by
- 如果x在A完成，y在B完成，则需额外割去x,y之间的边，花费ax+by+v
- 如果x在B完成，y在A完成，也需要割去x,y之间的边，花费bx+ay+v

可以发现，每一组方案都对应着一组割，而最小的割便是最优的方案了。通过本题，我们得到一种最小割的建模，并将它与二元组的关系一一对应起来，接下来我们将问题一般化，看看这个模型是否仍然能够胜任。

3 问题一般化

3.1 问题描述

有N个任务，每个任务可以在机器A或者机器B上完成，花费分别为a_i或者b_i。对于每一对二元组(x,y)，有如下关系：

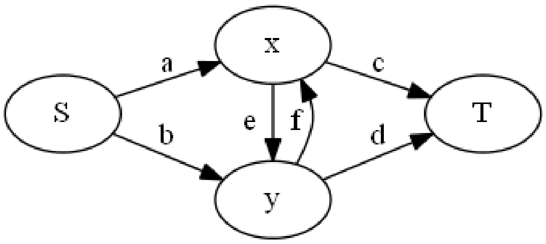
- 如果x与y同时在机器A上完成，增加花费v1[xy]
- 如果x与y同时在机器B上完成，增加花费v2[xy]
- 如果x在A机器上完成，y在B机器上完成，增加花费v3[xy]
- 如果x在B机器上完成，y在A机器上完成，增加花费v4[xy]

我们暂且不规定以上出现的所有数的符号。需要安排完成每个任务的机器使得总花费最小。

3.2 建造模型

首先，对于每个任务自身的花费ai或者bi，我们可以最后再加到割边上去，所以暂且不考虑。由于权值可加性，我们可以对于每一组关系分开考虑，最后再合并。

现在我们要做的，是对网络流中的每条边进行赋值，使得割的方案能够对应完成任务的方案。让我们先给这些边设上未知数：



它们需要满足的方程是：

$$a + b = v1 \tag{1}$$

$$c + d = v2 \tag{2}$$

$$a + d + f = v3 \tag{3}$$

$$b + c + e = v4 \tag{4}$$

注意不会存在类似$a + b + d$的方案，虽然它也是割，可是它显然没有$a + b$优。

让我们一步一步研究这个方程的性质：

1. 可以发现，在最小割问题中，每条边的边权都应该是非负的。因为出现负数边权，我们一定会把它割掉（这样才符合最小割的定义），可是这样做与我们题目要求的并不相符，因为不能做到方案一一对应了（会存在某些边一定被割）。所以我们需要最后解出的a, b, c, d, e, f均≥ 0。

2. 进一步思考，可以发现a与c我们选择且仅选择一个，b与d也是如此。如果$a, c < 0$，我们可以同时令a和c加上一个值，最后求出最小割后再将这个值减掉。于是即便$a, b, c, d < 0$也是可行的，上面的结论我们修改成$e, f \geq 0$。

3. 方程中有6个未知数，4个方程，说明很可能有多组解。

4. (3)式+(4)式-(1)式-(2)式，可得$f + e = v3 + v4 - v1 - v2$。我们令$v3 + v4 - v1 - v2 = K$，由于$f$和$e$必须$\geq 0$，所以$K$也必须$\geq 0$。

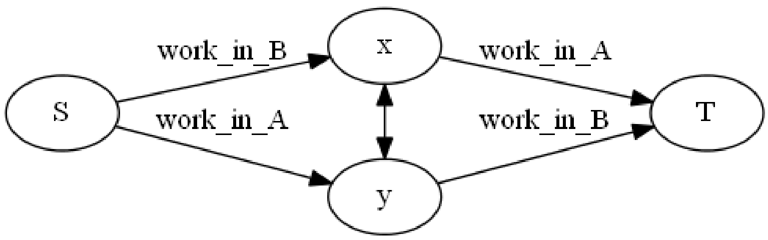
5. 由于a, b, c, d没有其他的限制条件，所以我们可以假定$f = e = \frac{K}{2}$，将两条单向边变为一条双向边，也减少了我们的计算量和思维量。

6. 最后a, b, c, d中我们任意确定一个数的值，则其他数的值也都可以解出（实际应用中往往选择看起来比较方便的值）。

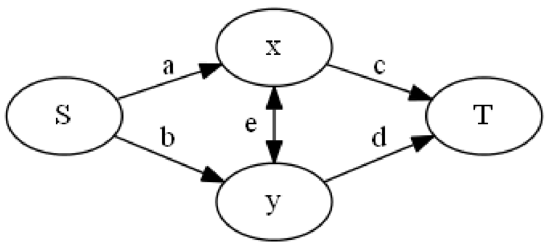
好了，到此为止，我们已经发现了一个解决二元关系的一个通用模型。唯一的遗憾是需要$K \geq 0$。

3.3 弥补缺憾

能不能弥补这个缺憾？我们仍然保留原来的模型，但是修改一下它的初始含义：若割掉S与x相连的边，表示在B机器上工作，若割掉x与T相连的边，表示在A机器上工作（和原来相反）。若割掉S与y相连的边，表示在A机器上工作，若割掉y与T相连的边，表示在B机器上工作（和原来相同）。即：



重复前面的工作，设未知数（注意由于f和e可以相等，所以我们舍去f）：



列出方程（注意方程发生的变化！）：

$$a + b = v4 \tag{5}$$

$$c + d = v3 \tag{6}$$

$$a + d + e = v2 \tag{7}$$

$$b + c + e = v1 \tag{8}$$

哈！我们惊讶地发现：$2e = v1 + v2 - v4 - v3 = -K$。也就是说，如果$K < 0$，我们可以通过这种方式，使得$K$变为$-K$从而满足$K \geq 0$。这真是令人欣喜的结论。不过要注意这种方法需要一个前提：若某一组二元组(x,y)的K值小于0，则x,y不能以同一种方式与S和T连接：将x与y连边，要求新建的图不能出现奇环，即该图是二分图。

也就是说，如果原图是二分图，我们就可以放心大胆地建模，否则就需要小心地去验证$K < 0$的关系构成的图是否存在奇环。

4 小试牛刀[happiness, 2010 吴确集训队测试题]

有了上面的模型，你一定很想找个题目练练手吧，下面就让我们一起来用该模型的思路解决一个问题。

4.1 题目描述

高一一班的座位表是个n*m的矩阵，经过一个学期的相处，每个同学和前后左右相邻的同学互相成为了好朋友。这学期要分文理科了，每个同学对于选择文科与理科有着自己的喜悦值，而一对好朋友如果能同时选文科或者理科，那么他们又将收获一些喜悦值。作为计算机竞赛教练的scp大老板，想知道如何分配可以使得全班的喜悦值总和最大。(喜悦值为正)

4.2 题目分析

4.2.1 最大->>最小

首先，题目要求喜悦值最大，我们先将给定的权值取负，使得问题变成总喜悦值最小。

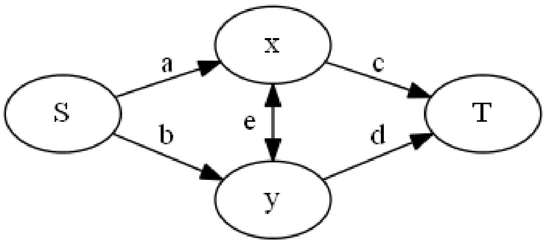
4.2.2 建立二元关系

每个同学本身对文理科的喜悦值可以最后加上。那么只需考虑下面两种情况：

1. 两个好朋友同时选文科，获得-v1的喜悦值

2. 两个好朋友同时选理科，获得-v2的喜悦值

4.2.3 设未知数和解方程



$$a + b = -v1 \tag{9}$$

$$c + d = -v2 \tag{10}$$

$$a + d + e = 0 \tag{11}$$

$$b + c + e = 0 \tag{12}$$

由于$K = 0 + 0 + v1 + v2 \geq 0$。于是我们无需担心二分图的问题。接下来只需解方程：

1. $e = K/2 = (v1 + v2)/2$
2. 设$a = -v1/2$，可得$b = -v1/2, c = -v2/2, d = -v2/2$。
3. 由于$a, b, c, d \leq 0$，所以我们令它们同时加上$(v1 + v2)/2$，化简得：$a = v2/2, b = v2/2, c = v1/2, d = v1/2$。最后再减去(v1 + v2)即可。
4. 再将整个权值乘以2，$a = b = v2, c = d = v1, e = v1 + v2$，使权值更为简洁

4.2.4 建图，求最小割

4.2.5 得出答案

求出最小割，再把我们中途做的变换逆回去即可得出答案。

4.2.6 注意

如果本题中不是喜悦值最大，而是总花费（代价）最小，则我们可以利用本题的二分图性质建图。

5 最大权闭合子图

子曰：学而时习之，不亦说乎。让我们将二元组关系模型和以前解决过的问题进行联系，看看是否也能得到不一样的乐趣。

5.1 问题描述

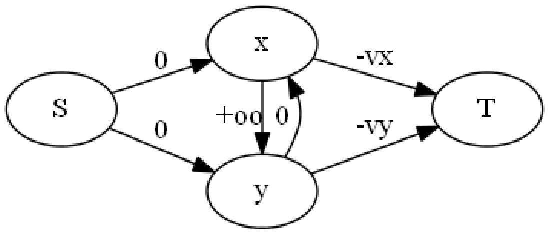
在一个有向无环图中，每个点都有点权（可正可负），现在需要选出一个子图，满足若一个点被选，则它所有的后继也要被选。使得该子图的点权和最大。

5.2 经典建模

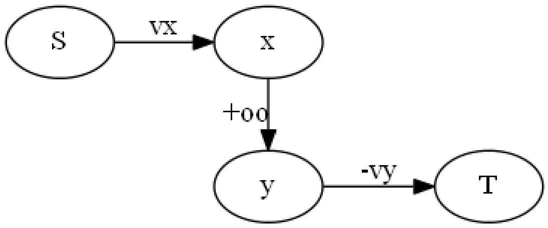
对于该类问题的经典做法是：源向所有的正权点连边，容量为正权，所有的负权点向汇连边，容量为负权的绝对值，原图的边保留，容量为正无穷。求出最小割后，用所有的正权点的权值和-最小割，即为答案。（在此不做证明）

5.3 二元组关系模型

1. 将最大值变为最小值，权值取负。
2. 确定二元关系：对于一条有向边(x,y)，若x选，则y必须选。换言之，x选，y不选，则需要付出正无穷的代价。
3. 定义x和T连的边表示选，S和x连的边表示不选。
4. 解方程，可以发现将e和f分开考虑更好。建出的图形（加上了每个点的点权）如下:



5. 假设vx是正权，我们将x两侧的点同时加上vx，假设vy是负权，则我们无需改变，化简以后的图如下：



怎么样？是不是和经典模型的建图有异曲同工之妙！可以发现，最大权闭合子图正是二元组关系的一种特殊形式。

6 最大权密度子图

让我们再来看看最大权密度子图和二元组建模的联系。

6.1 问题描述

在一个带点权带边权的无向图中，我们需要选出一张子图，满足该子图的点权和比上该子图的边权和最大。

6.2 二元组关系建模

1. 首先二分答案k，问题转换为$|V| - k|E|$最大。

2. 定义二元关系：如果一条边连接的两个点同时被选，则将获得该边的权值（可能需要考虑其正负情况）。

3. 解方程，求最小割，So easy!。

7 Topcoder srm 558 div1 t3

接下来让我们看一道不是那么经典的问题。

7.1 题目描述

有一个n*m的方阵，每个格子可以放石头

- 如果一个格子放了石头，则花费c
- 如果一个格子放了石头，或者它的周围的格子都放了石头，则收益v

问怎么放石头，使得收益-花费的值最大。（c，v皆为正数）

7.2 题目分析

本题的关键字是："如果周围的格子**都**放了石头，收益v"。这似乎在告诉我们本题已经不是二元关系了，而是一对多元。没有关系，让我们新增一个点x1，表示周围的格子是否都放了石头，就可以让它回归到二元关系的行列。

x的状态选择是放还是不放，而x1的状态选择是"周围有没有都放"，它们的选择本身会带来代价。

首先考虑x与x1之间的二元关系：

- 如果x选放且x1也选放，则失去v的收益

然后考虑x1与(x的相邻格子)y之间的二元关系

- 如果x1选，则y必须要选，如果y不选，则将花费正无穷的代价

依据上述二元关系列方程，再进行建图（本题需要二分图关系）即可解决。

8 文理分科，2013湖南省队集训，彭天翼

8.1 题目描述

在一个n*m的网格图中，每一个点需要被赋予0或者1两种属性。获得的价值按如下的方式计算：

- 第i行第j列的格子为0，获得a_{ij}的价值。
- 第i行第j列的格子为1，获得b_{ij}的价值。
- 第i行第j列的格子为0，且与其相邻的格子都为0，则增加Sa_{ij}的价值。
- 第i行第j列的格子为1，且与其相邻的格子都为1，则增加Sb_{ij}的价值。

问怎样安排使得价值最大。

8.2 题目分析

这是我基于topcoder 558 div1 t3而出的一道加强题。做法和上题类似，新增两个点x1,x2，分别表示周围的格子是否都为0及周围的格子是否都为1。

二元关系有：

- 如果x为0且x1为是，则增加Sa的值。
- 如果x为1且x2为是，则增加Sb的值。
- 如果x1为是，则y必须为0。
- 如果x2为是，则y必须为1。

依据二元关系解方程，建图即可。

9 总结

9.1 模型适用范围

有N个元素，每个元素在两种状态中选且仅选择一个，选择可以有代价。有二元组关系，描述两个元素之间不同关系时产生的代价。问怎么选择每个元素的状态，使得总代价和最小。在特殊情况下：【$K < 0$】需要二分图的条件。

9.2 解题关键

找出题目中的二元关系。

9.3 需要注意的地方

在转换的过程中，涉及的步骤有点多【比如最大变最小，边权同时加上值，边权同时乘以一个值】，需要步步为营，小心谨慎。

10 另一份总结

在总结之前，我想先说一说我自己的一点经历：

在小学的时候，我接触到很多的基础奥赛知识，比如鸡兔同笼问题，追及相遇问题，盈亏问题，这些有趣的问题伴我走过整个小学，我常常为弄懂了它们的解法而感到开心。直到初中我学会了设未知数与解方程，我才发现："噢！原来它们都可以用同一种方法解决！"，"噢！原来它们的通解是这么的简单！"。这种妙不可言的感觉伴随了我很久。回想起来，我并不以为小学时的千奇百怪的解法是在浪费时间，它们同样给了我带来了很大的启发和快乐。

在这里，我想把我们以前学会的某些最小割模型（最大密度子图等）比作鸡兔同笼问题，而本篇论文所述的二元组模型比作设未知数与解方程（可能这个比喻并不恰当）。当我们学会了这个更一般的解法后，有助于我们对该类问题有更深刻的理解，在解决此类问题时可以更省心一些。当然，以前的做法和证明方法同样对我们也有裨益。

大而泛谈地说，很可能我们人类现在所学习的，所掌握的也只不过是一些更为基础的问题的特殊形式，特殊解法。可能很多我们现在看起来毫不相干的东西，其实本质相同（就好像以前的人看石墨和金刚石（都由碳元素构成）一般）。在追求世界万物的统一，探寻自然运转规律的道路上，人类越走越远，可是这条道路，却不一定会有尽头。。咳，扯远了。。

11 补充

本篇论文行文仓促，如果有对题目解法提出异议或询问，或者对本模型有着其它想法和应用的同学，欢迎和我进行交流和探讨。我的邮箱是：380759124@qq.com。

12 致谢

感谢胡伯涛同学精彩又经典的论文对我的启蒙和启发。

感谢魏铭同学在北京时的精彩讲课和课件对我的帮助。

感谢战友钟泽轩同学，谢天成同学，胡泽聪同学与我一起进行学术上的探讨和思考以及干过的各种有趣的事。

感谢教练李淑平老师对我无微不至的帮助和照顾。

感谢竞赛路上曾给予我帮助的同学和老师。

参考文献

[1] 胡伯涛，《最小割模型在信息学中的应用》

[2] 魏铭，《浅谈信息学中的网络流问题》

[3] 吴确，《2010年集训队测试题》